APPENDIX 1

(A Script)

------------ Generate new script-------------

0=generateStandardPreamble

1=generateText(Java EJB imports,Common.ini,random,vendor.csv)

2=generateText(com.testmybeans.vendor.Vendor imports,ctor.ini,random,vendor.csv)

3=generateText(TestThread header,Common.ini,random,vendor.csv)

4=generateText(TestThread.run implementation,Common.ini,random,vendor.csv)

5=generateText(com.testmybeans.vendor.Vendor create,com.testmybeans.vendor.Vendor.ctor.ini,random,vendor.csv)

6=generateMethod(timingGetSet)

7=generateText(Bean end create,Common.ini,random,vendor.csv)

8=generateText(WebLogic getInitialContext,AppServer.ini,random,vendor.csv)

9=generateText(Close log,Common.ini,random,vendor.csv)

10=generateText(Log to disk,Common.ini,random,vendor.csv)

11=generateStandardTrailer

```
      [Java EJB imports]
      //[Java EJB imports]
5     package com.testmybeans.client;
      import javax.naming.InitialContext;
      import javax.naming.Context;
      import javax.naming.NamingException;
      import java.rmi.RemoteException;
10    import java.util.*;
      import java.io.*;


      [Hello imports]
      //[Hello imports]
15    import com.softbridge.hello.HelloHome;
      import com.softbridge.hello.Hello;
      import com.softbridge.hello.HelloPK;


      [TestThread header]
20    //[TestThread header]
      public class $system.programName$ extends Thread
      {
      int tempInt=0;
      Double tempDouble=new Double(0.0);
25    Float tempFloat=new Float(0.0);
      short tempShort=0;
      long tempLong=0;
      String tempString="";
      long sTime=0;
30    long sStartTime=0;
      int m_instanceNumber = 0;
      static String m_urlName = null;
      String threadNumber="";
      public $system.programName$(int inst, String url)
35    {
      m_instanceNumber = inst;
```

```
        m_urlName = url;
        threadNumber=Integer.toString(m_instanceNumber);
        //System.out.println("Starting Instance: " + m_instanceNumber);
        }

5

        [TestThread.run implementation]
        //[TestThread.run implementation]
        public void run()
        {
10      sTime = new Date().getTime();
        sStartTime=sTime;
        logError("elapsedStart", Long.toString(new Date().getTime()), "", "");


        [Bean end create]
15      //[Bean end create]
        }
        catch (Exception e)
        {
        logError("except", "beanMethods", e.toString(), "");
20      }
        finally
        {
        try
        {
25      closeLog();
        h.remove ();
        h = null;
        }
        catch (Exception e)
30      {
        logError("except", "closeHome", e.toString(), "");
        }
        }
        }
35

        [Bean end findByPrimaryKey]
```

```
//[Bean end findByPrimaryKey]
}
catch (Exception e)
{
logError("except", "beanMethods", e.toString(), "");
}
finally
{
try
{
closeLog();
h = null;
}
catch (Exception e)
{
logError("except", "closeHome", e.toString(), "");
}
}
}


[Log to disk]
//[Log to disk]
public void logError(String key, String functionName, String expected, String actual)
{
try
{
String elapsedTime = Long.toString(new Date().getTime() - sTime);
com.testmybeans.client.CThreadWriter.write(key + "~" + functionName + "~" + expected
+ "~" + actual + "~" + elapsedTime + "~" + threadNumber);
}
catch (Exception e)
{
System.out.println("IOException: " + e.getMessage());
System.exit(1);
}
}
```

```
          }

          [Close log]
          //[Close log]
5         public void closeLog()
          {
          sTime=sStartTime;
          logError("elapsedEnd", Long.toString(new Date().getTime()), "", "");
          }
10
```

```
//
//Bean under test: com.testmybeans.vendor.Vendor
//Author: TestMyBeans Code Generator v0.8
//Creation Date: Fri Dec 10 16:53:15 EST 1999
//Copyright (c) 1999 TestMyBeans, Inc. All rights reserved.
//
//[Java EJB imports]
package com.testmybeans.client;
import javax.naming.InitialContext;
import javax.naming.Context;
import javax.naming.NamingException;
import java.rmi.RemoteException;
import java.util.*;
import java.io.*;


//[TestThread header]
public class invoice extends Thread
    {
    int tempInt=0;
    Double tempDouble=new Double(0.0);
    Float tempFloat=new Float(0.0);
    short tempShort=0;
    long tempLong=0;
    String tempString="";
    long sTime=0;
    long sStartTime=0;
    int m_instanceNumber = 0;
    static String m_urlName = null;
    String threadNumber="";
    public invoice(int inst, String url)
        {
        m_instanceNumber = inst;
        m_urlName = url;
```

```java
        threadNumber=Integer.toString(m_instanceNumber);
        //System.out.println("Starting Instance: " + m_instanceNumber);
        }


5      //[TestThread.run implementation]
        public void run()
            {
        sTime = new Date().getTime();
        sStartTime=sTime;
10      logError("elapsedStart", Long.toString(new Date().getTime()), "", "");


        //[com.testmybeans.vendor.Vendor create]
        int msValue = (-100000) * ((int) new Date().getTime ()) %
        (Integer.MAX_VALUE/100000);
15      int arg0 =  m_instanceNumber  + msValue;
        com.testmybeans.vendor.Vendor h = null;
        try
            {
            Context jndi = getInitialContext();
20          com.testmybeans.vendor.VendorHome home =
        (com.testmybeans.vendor.VendorHome)jndi.lookup("OEVendor");
            h = home.create(arg0);
            }
        catch (Exception e)
25          {
            System.out.println ("com.testmybeans.vendor.VendorHome" + e.toString());
            }
        try
            {
30          //-----------com.testmybeans.vendor.Vendor(setCost) -------------
            try
                {
            sTime = new Date().getTime();
            h.setCost(568);
35          logError("SetTime", "setCost", "568", "");
                }
```

- 48 -

```
        catch (Exception e)
            {
            logError("except", "setCost", e.toString(), "");
            }
5   //-----------com.testmybeans.vendor.Vendor(setName) -------------
    try
            {
            sTime = new Date().getTime();
            h.setName("qkgthfpw");
10          logError("setTime", "setName", "qkgthfpw", "");
            }
    catch (Exception e)
            {
            logError("except", "setName", e.toString(), "");
15          }
    //-----------com.testmybeans.vendor.Vendor(setProduct) -------------
    try
            {
            sTime = new Date().getTime();
20          h.setProduct("dpnoolww");
            logError("setTime", "setProduct", "dpnoolww", "");
            }
    catch (Exception e)
            {
25          logError("except", "setProduct", e.toString(), "");
            }
    //-----------com.testmybeans.vendor.Vendor(getCost) -------------
    try
            {
30          sTime = new Date().getTime();
            if((tempInt = h.getCost()) != 555)
            logError("getFailed", "getCost", "555", Integer.toString(tempInt));
            else
            logError("getPassed", "getCost", "555", Integer.toString(tempInt));
35          }
    catch (Exception e)
```

```java
            {
            logError("except", "getCost", e.toString(), "");
            }
        //-----------com.testmybeans.vendor.Vendor(getName) -------------
        try
            {
            sTime = new Date().getTime();
            if(!(tempString = h.getName()).equals("icgnhoje"))
            logError("getFailed", "getName", "icgnhoje", tempString);
            else
            logError("getPassed", "getName", "icgnhoje", tempString);
            }
        catch (Exception e)
            {
            logError("except", "getName", e.toString(), "");
            }
        //-----------com.testmybeans.vendor.Vendor(getProduct) -------------
        try
            {
            sTime = new Date().getTime();
            if(!(tempString = h.getProduct()).equals("zwlhjoxk"))
            logError("getFailed", "getProduct", "zwlhjoxk", tempString);
            else
            logError("getPassed", "getProduct", "zwlhjoxk", tempString);
            }
        catch (Exception e)
            {
            logError("except", "getProduct", e.toString(), "");
            }
        //[Bean end create]
        }
    catch (Exception e)
        {
        logError("except", "beanMethods", e.toString(), "");
        }
    finally
```

- 50 -

```java
        {
        try
            {
            closeLog();
            h.remove ();
            h = null;
            }
        catch (Exception e)
            {
            logError("except", "closeHome", e.toString(), "");
            }
        }
    }

//[WebLogic getInitialContext]
public static Context getInitialContext() throws javax.naming.NamingException
    {
    try
        {
        Properties p = new Properties();
        p.put(Context.INITIAL_CONTEXT_FACTORY,
"weblogic.jndi.T3InitialContextFactory");
        p.put(Context.PROVIDER_URL, m_urlName);
        return new javax.naming.InitialContext(p);
        }
    catch (Exception e)
        {
        System.out.println("getInitialContext Exception: "+e);
        System.exit(1);
        }
    return null;
    }
//[Close log]
public void closeLog()
        {
        sTime=sStartTime;
```

```java
        logError("elapsedEnd", Long.toString(new Date().getTime()), "", "");
        }


    //[Log to disk]
    public void logError(String key, String functionName, String expected, String actual)
        {
        try
            {
            String elapsedTime = Long.toString(new Date().getTime() - sTime);
            com.testmybeans.client.CThreadWriter.write(key + "~" + functionName + "~" +
expected + "~" + actual + "~" + elapsedTime + "~" + threadNumber);
            }
        catch (Exception e)
            {
            System.out.println("IOException: " + e.getMessage());
            System.exit(1);
            }
        }
        }
    //
    //Bean under test: com.testmybeans.vendor.Vendor
    //Author: TestMyBeans Code Generator v0.8
    //Creation Date: Fri Dec 10 16:53:15 EST 1999
    //Copyright (c) 1999 TestMyBeans, Inc. All rights reserved.
    //
    //[Java EJB imports]
    package com.testmybeans.client;
    import javax.naming.InitialContext;
    import javax.naming.Context;
    import javax.naming.NamingException;
    import java.rmi.RemoteException;
    import java.util.*;
    import java.io.*;



    //[TestThread header]
```

```
public class invoice extends Thread
    {
    int tempInt=0;
    Double tempDouble=new Double(0.0);
    Float tempFloat=new Float(0.0);
    short tempShort=0;
    long tempLong=0;
    String tempString="";
    long sTime=0;
    long sStartTime=0;
    int m_instanceNumber = 0;
    static String m_urlName = null;
    String threadNumber="";
    public invoice(int inst, String url)
        {
        m_instanceNumber = inst;
        m_urlName = url;
        threadNumber=Integer.toString(m_instanceNumber);
        //System.out.println("Starting Instance: " + m_instanceNumber);
        }

    //[TestThread.run implementation]
    public void run()
        {
        sTime = new Date().getTime();
        sStartTime=sTime;
        logError("elapsedStart", Long.toString(new Date().getTime()), "", "");

        //[com.testmybeans.vendor.Vendor create]
        int msValue = (-100000) * ((int) new Date ().getTime ()) %
(Integer.MAX_VALUE/100000);
        int arg0 =  m_instanceNumber  + msValue;
        com.testmybeans.vendor.Vendor h = null;
        try
            {
            Context jndi = getInitialContext();
```

```
            com.testmybeans.vendor.VendorHome home =
(com.testmybeans.vendor.VendorHome) jndi.lookup("OEVendor");
            h = home.create(arg0);
            }
    catch (Exception e)
            {
            System.out.println ("com.testmybeans.vendor.VendorHome" + e.toString());
            }
        try
            {
            //-----------com.testmybeans.vendor.Vendor(setCost) -------------
            try
                {
                sTime = new Date().getTime();
                h.setCost(568);
                logError("SetTime", "setCost", "568", "");
                }
            catch (Exception e)
                {
                logError("except", "setCost", e.toString(), "");
                }
            //-----------com.testmybeans.vendor.Vendor(setName) -------------
            try
                {
                sTime = new Date().getTime();
                h.setName("qkgthfpw");
                logError("setTime", "setName", "qkgthfpw", "");
                }
            catch (Exception e)
                {
                logError("except", "setName", e.toString(), "");
                }
            //-----------com.testmybeans.vendor.Vendor(setProduct) -------------
            try
                {
                sTime = new Date().getTime();
```

- 54 -

```
            h.setProduct("dpnoolww");
            logError("setTime", "setProduct", "dpnoolww", "");
            }
        catch (Exception e)
5           {
            logError("except", "setProduct", e.toString(), "");
            }
        //-----------com.testmybeans.vendor.Vendor(getCost) ------------
        try
10          {
            sTime = new Date().getTime();
            if((tempInt = h.getCost()) != 555)
            logError("getFailed", "getCost", "555", Integer.toString(tempInt));
            else
15          logError("getPassed", "getCost", "555", Integer.toString(tempInt));
            }
        catch (Exception e)
            {
            logError("except", "getCost", e.toString(), "");
20          }
        //-----------com.testmybeans.vendor.Vendor(getName) ------------
        try
            {
            sTime = new Date().getTime();
25          if(!(tempString = h.getName()).equals("icgnhoje"))
            logError("getFailed", "getName", "icgnhoje", tempString);
            else
            logError("getPassed", "getName", "icgnhoje", tempString);
            }
30      catch (Exception e)
            {
            logError("except", "getName", e.toString(), "");
            }
        //-----------com.testmybeans.vendor.Vendor(getProduct) ------------
35      try
            {
```

```java
                sTime = new Date().getTime();
                if(!(tempString = h.getProduct()).equals("zwlhjoxk"))
                logError("getFailed", "getProduct", "zwlhjoxk", tempString);
                else
5               logError("getPassed", "getProduct", "zwlhjoxk", tempString);
                }
            catch (Exception e)
                {
                logError("except", "getProduct", e.toString(), "");
10              }
            //[Bean end create]
            }
        catch (Exception e)
            {
15          logError("except", "beanMethods", e.toString(), "");
            }
        finally
            {
            try
20              {
                closeLog();
                h.remove ();
                h = null;
                }
25          catch (Exception e)
                {
                logError("except", "closeHome", e.toString(), "");
                }
            }
30      }


    //[WebLogic getInitialContext]
    public static Context getInitialContext() throws javax.naming.NamingException
        {
35      try
            {
```

```java
        Properties p = new Properties();
        p.put(Context.INITIAL_CONTEXT_FACTORY,
"weblogic.jndi.T3InitialContextFactory");
        p.put(Context.PROVIDER_URL, m_urlName);
        return new javax.naming.InitialContext(p);
        }
    catch (Exception e)
        {
        System.out.println("getInitialContext Exception: "+e);
        System.exit(1);
        }
    return null;
        }
    //[Close log]
    public void closeLog()
        {
        sTime=sStartTime;
        logError("elapsedEnd", Long.toString(new Date().getTime()), "", "");
        }

    //[Log to disk]
    public void logError(String key, String functionName, String expected, String actual)
        {
        try
        {
        String elapsedTime = Long.toString(new Date().getTime() - sTime);
        com.testmybeans.client.CThreadWriter.write(key + "~" + functionName + "~" +
expected + "~" + actual + "~" + elapsedTime + "~" + threadNumber);
        }
    catch (Exception e)
        {
        System.out.println("IOException: " + e.getMessage());
        System.exit(1);
        }
        }
    }
```